

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

Welcome to IEEE Xplore®

- Home
- What Can I Access?
- Log-out

Tables of Contents

- Journals & Magazines
- Conference Proceedings
- Standards

Search

- By Author
- Basic
- Advanced

Member Services

- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

 **Print Format**

[Home](#) | [Log-out](#) | [Journals](#) | [Conference Proceedings](#) | [Standards](#) | [Search by Author](#) | [Basic Search](#) | [Advanced Search](#) | [Join IEEE](#) | [Web Account](#) | [New this week](#) | [OPAC Linking Information](#) | [Your Feedback](#) | [Technical Support](#) | [Email Alerting](#) | [No Robots Please](#) | [Release Notes](#) | [IEEE Online Publications](#) | [Help](#) | [FAQ](#) | [Terms](#) | [Back to Top](#)

Copyright © 2004 IEEE — All rights reserved

Searching for **graphical programming and remote execute**.

Restrict to: [Header](#) [Title](#) [Order by: Expected citations](#) [Hubs](#) [Usage](#) [Date](#) Try: [Amazon](#) [B&N](#) [Google](#)
(CiteSeer) [Google \(Web\)](#) [CSB](#) [DBLP](#)

No documents match Boolean query. Trying non-Boolean relevance query.

1000 documents found. Retrieving documents... Order: relevance to query.

[Designing Parallel Programs by the Graphical Language GRAPNEL - Eter Kacsuk \(1996\) \(Correct\) \(11 citations\)](#)

Designing Parallel Programs by the **Graphical Language GRAPNEL** 3 P' eter Kacsuk, G' abor D'

Abstract We propose a new visual **programming** language, called GRAPNEL (**GR**aphical Process's

www.kfki.hu/~mszkihp/info/ParComp/papers/EuroMicroPSE-grapnel.ps.Z

[Talking Vs Taking: Speech Access To Remote Computers - Yankelovich \(1994\) \(Correct\) \(2 citations\)](#)

we did not attempt an exact translation of the **graphical** interface. We set about designing the speech

Talking Vs Taking: Speech Access To Remote Computers Nicole Yankelovich Sun Microsystems

www.sunlabs.com/research/speech/publications/chi94/CHI94Short.ps

[An Initial Assessment of Aspect-oriented Programming - Robert Walker \(1998\) \(Correct\) \(6 citations\)](#)

An Initial Assessment of Aspect-oriented **Programming** Robert J. Walker, Elisa L.A. Baniassad and

www.cs.ubc.ca/spider/walker/papers/walker.1998c.ps.gz

[On the Utility of Threads for Data Parallel Programming - Fahringer, Haines, Mehrotra \(1995\) \(Correct\) \(4 citations\)](#)

On the Utility of Threads for Data Parallel **Programming** Thomas Fahringer y Matthew Haines z

separating the access times for local and **remote** memories. Therefore, it is highly desirable to independent, sequential unit of computation that **executes** within the context of a kernel-supported entity,

ftp.ics.uci.edu/pub/techreports/95/95-35.ps.Z

[The Nexus Approach to Integrating Multithreading and Communication - Foster \(1996\) \(Correct\) \(123 citations\)](#)

problems and algorithms. In these situations, the **programmer** can reasonably be assumed to have global mask communication and I/O latencies, to implement **remote** memory access, and to support task-parallel and operations be decoupled? Should the code **executed** in response to an incoming message be specified now.cs.berkeley.edu/clumps/jpdc.ps

[The Design of a Completely Visual Object-Oriented.. - Citrin, Doherty, Zorn \(1994\) \(Correct\) \(4 citations\)](#)

our language can be entirely described by simple **graphical** rules. Vipr provides a framework for and

To appear in **Visual Object-Oriented Programming: Concepts and Environments**, Margaret Burnett,

ftp.cs.colorado.edu/pub/techreports/zorn/VOOP-VIPR.ps.Z

[The MViews framework for constructing multi-view editing.. - Grundy, Hosking \(1993\) \(Correct\)](#)

environments that support integrated textual and **graphical programming** with consistency management. It

that support integrated textual and **graphical programming** with consistency management. It provides a

www.cs.waikato.ac.nz/~jgrundt/papers/nzjc93.ps.Z

[A Hypertext System for Integrating Heterogeneous, Autonomous.. - Noll, Scacchi \(1994\) \(Correct\) \(2 citations\)](#)

with the structured and semi-structured (textual, **graphical**) data prevalent in software environments. 2)

database software will use the traditional Unix **programming** tools (cc, make, RCS) to develop and test

location transparency, the ability to access **remote** objects in the same manner as local objects and

cwis.usc.edu/dept/ATRIUM/Papers/Integrating_Software_Repositories.ps

[Nets, Logic and Concurrent Object-Oriented Programming - Rafael Ramirez \(Correct\)](#)

are linked by an infinite buffer. We can **graphically** represent the behaviour of the producer

Nets, logic and concurrent object-oriented **programming** Rafael Ramirez Dept. of Computer Science

interpretation which allows specifications to be **executed** as concurrent programs. Thus, the execution of

www.cs.bris.ac.uk/~ramirez/Petri.ps

[Cspack Client-Server Routines And Utilities - Cern \(Correct\)](#)

manner. Not only is the user able to display **graphical** output from the **remote** session in a window on

:4 1.2.2 XZ -The **remote** I/O package :

wwwinfo.cern.ch/asdoc./psdir/cspack.ps.gz

A Graphical Programming Environment for Message Passing.. - Kacsuk, Dózsa.. (1997) (Correct) (1 citation)

A Graphical Programming Environment for Message Passing

A Graphical Programming Environment for Message Passing Programs P.

<ftp://lpds.sztaki.hu/pub/lpds/publications/GRADE/grade-pdse97.ps.gz>

Shared Memory NUMA Programming on I-WAY - Nieplocha, Harrison (1996) (Correct) (8 citations)

1 Shared Memory NUMA Programming on I-WAY J. Nieplocha and R. J. Harrison

of mirrored arrays. Latencies and bandwidths for **remote** memory access are studied, and the performance of which on distributed-memory architectures **executes** on a dedicated processor, that is devoted

<ftp://pnl.gov/pub/permanent/global/iway.ps.Z>

PVMGraph: A Graphical Editor For the Design of PVM Programs - Justo (1996) (Correct) (7 citations)

No: EDPEPPS/5 Deliverables: 2.3.3 PVMGraph: A Graphical Editor For the Design of PVM Programs G. R.

<ftp://cpc.wmin.ac.uk/pub/HPCTI/P4/appendix-G1.ps.gz>

David G. Goodenough - Daniel Charlebois (Correct)

a product used to make land use decisions via a **graphical** user interface. A product can be a digital information file into an image format or ingest **remote** sensing data and update meta data databases. In reasoning system to assemble plans that can be **executed** to create integrated products [1,2]SEIDAM: www.engr.uvic.ca/~ndaley/nigel/inv_update.ps

Merging Interactive, Modular, And Object-Oriented Programming - Tung (Correct)

paradigm. Examples of these applications include **graphical** user interfaces, data base systems,

Merging Interactive, Modular, And Object-Oriented Programming Sho-Huan Simon Tung Submitted To The Faculty <ftp://cs.indiana.edu/pub/techreports/TR349.ps.Z>

The Design and Implementation of Arjuna - Parrington, al. (1995) (Correct) (34 citations)

Tyne, Ne1 7ru Uk Arjuna Is An Object-Oriented **Programming** System Implemented Entirely In CThat services for interprocess communication (e.g.**remote** procedure calls)naming and binding (for being the assumption that the application will **execute** within a single address space. Secondly, support www.fcul.research.ec.org/broadcast/trs./papers/65.ps

Tcl/Tk for a Personal Digital Assistant - Karin Petersen (Correct)

level language which can be used to program the **graphical** user interfaces for the PDA. In addition, the common infrastructure for PDA user interface **programming** and as the communication platform between the design and communication between the PDA and **remote** applications. The result was that Tcl/Tk enabled sandbox.parc.xerox.com/petersen/vhll.ps.Z

Using Remote Memory to avoid Disk Thrashing: A Simulation Study - Markatos (1996) (Correct) (1 citation)

In Proceedings of the SIGPLAN '94 Conference on **Programming** Language Design and Implementation, Orlando,

Using **Remote** Memory to avoid Disk Thrashing: A Simulation

simulation routines. The application is actually **executed**, while at the same time, our simulator runs to

www.ics.forth.gr/proj/avg/papers/1996.MASCOTS96.Remote_memory_paging.ps.gz

Constructing Multi-View Editing Environments Using MViews - Grundy, Hosking (1993) (Correct) (2 citations)

environments that support integrated textual and **graphical programming**. It provides a conceptual model and that support integrated textual and **graphical programming**. It provides a conceptual model and reusable

[ftp://ftp.cs.waikato.ac.nz/pub/papers/mviews/VL93.ps.gz](http://ftp.cs.waikato.ac.nz/pub/papers/mviews/VL93.ps.gz)

HeNCE: A Heterogeneous Network Computing Environment - Beguelin, Dongarra, Geist, .. (1993) (Correct)

(19 citations)

Network Computing Environment) is an integrated **graphical** environment for creating and running parallel called PVM. The HeNCE philosophy of parallel **programming** is to have the **programmer** **graphically** specify uses a program called pvmrsh for carrying out the **remote** makes. This global make handles cases when the www.netlib.org/ncwn/hence-sp.ps

[First 20 documents](#) [Next 20](#)

Try your query at: [Amazon](#) [Barnes & Noble](#) [Google \(CiteSeer\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)



US Patent & Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: The ACM Digital Library The Guide

graphical programming remote execute panel



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used [graphical](#) [programming](#) [remote](#) [execute](#) [panel](#)

Found 387 of 134,837

Sort results
by

relevance

Save results to a Binder

Try an [Advanced Search](#)

Display
results

expanded form

Search Tips
 Open results in a new window

Try this search in [The ACM Guide](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale

1 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available: [pdf\(4.21 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

2 [A control panel interface for graphics and image processing applications](#)

Gene L. Fisher, Kenneth I. Joy

May 1986 **ACM SIGCHI Bulletin, Proceedings of the SIGCHI/GI conference on Human factors in computing systems and graphics interface**, Volume 17 Issue SI

Full text available: [pdf\(562.50 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper describes a graphical interface for application programs. The interface is based on the notion of a control panel. A control panel contains a browsable list of an application's parameters and a set of functions to control the application's execution. A variety of graphical knobs and gauges may be associated with any or all of the parameters to permit fine-grain execution control, including animation of an application's output. The control panel interface is integ ...

Keywords: applications environment, graphical interface, graphics applications, image processing applications

3 [A structural view of the Cedar programming environment](#)

Daniel C. Swinehart, Polle T. Zellweger, Richard J. Beach, Robert B. Hagmann

August 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 8 Issue 4

Full text available: [pdf\(6.32 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper presents an overview of the Cedar programming environment, focusing on its

overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. T ...

4 Paralex: an environment for parallel programming in distributed systems

Özalp Babaoglu, Lorenzo Alvisi, Alessandro Amoroso, Renzo Davoli, Luigi Alberto Giachini
August 1992 **Proceedings of the 6th international conference on Supercomputing**

Full text available:  [pdf\(1.07 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Modern distributed systems consisting of powerful workstations and high-speed interconnection networks are an economical alternative to special-purpose super computers. The technical issues that need to be addressed in exploiting the parallelism inherent in a distributed system include heterogeneity, high-latency communication, fault tolerance and dynamic load balancing. Current software systems for parallel programming provide little or no automatic support towards these issues and require ...

5 Programming languages and systems for prototyping concurrent applications

Wilhelm Hasselbring
March 2000 **ACM Computing Surveys (CSUR)**, Volume 32 Issue 1

Full text available:  [pdf\(559.78 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

Concurrent programming is conceptually harder to undertake and to understand than sequential programming, because a programmer has to manage the coexistence and coordination of multiple concurrent activities. To alleviate this task several high-level approaches to concurrent programming have been developed. For some high-level programming approaches, prototyping for facilitating early evaluation of new ideas is a central goal. Prototyping is used to explore the ...

Keywords: concurrency, distribution, parallelism, rapid prototyping, very high-level languages

6 The design, implementation, and evaluation of Jade

Martin C. Rinard, Monica S. Lam
May 1998 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 20 Issue 3

Full text available:  [pdf\(576.88 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Jade is a portable, implicitly parallel language designed for exploiting task-level concurrency. Jade programmers start with a program written in a standard serial, imperative language, then use Jade constructs to declare how parts of the program access data. The Jade implementation uses this data access information to automatically extract the concurrency and map the application onto the machine at hand. The resulting parallel execution preserves the semantics of the original serial program ...

Keywords: parallel computing, parallel programming languages

7 A Survey of Interactive Graphical Systems for Mathematics

Lyle B. Smith
December 1970 **ACM Computing Surveys (CSUR)**, Volume 2 Issue 4

Full text available: [pdf\(5.05 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

8 Java resources for computer science instruction

Joseph Bergin, Thomas L. Naps, Constance G. Bland, Stephen J. Hartley, Mark A. Holliday, Pamela B. Lawhead, John Lewis, Myles F. McNally, Christopher H. Nevison, Cheng Ng, George J. Pothering, Tommi Teräsvirta

December 1998 **Working Group reports of the 3rd annual SIGCSE/SIGCUE ITiCSE conference on Integrating technology into computer science education**

Full text available: [pdf\(107.98 KB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

9 Java resources for computer science instruction

Joseph Bergin, Thomas L. Naps, Constance G. Bland, Stephen J. Hartley, Mark A. Holliday, Pamela B. Lawhead, John Lewis, Myles F. McNally, Christopher H. Nevison, Cheng Ng, George J. Pothering, Tommi Teräsvirta

October 1998 **ACM SIGCUE Outlook**, Volume 26 Issue 4

Full text available: [pdf\(2.23 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The goal of this working group was to collect, evaluate, and foster the development of resources to serve as components of both new and revised traditional courses that emphasize object-oriented software development using Java. These courses could, for example, integrate Internet-based distributed programming, concurrency, database programming, graphics and visualization, human interface design and object-oriented development. They could therefore also be suitable as capstone courses in computer ...

10 A distributed 3D graphics library

Blair MacIntyre, Steven Feiner

July 1998 **Proceedings of the 25th annual conference on Computer graphics and interactive techniques**

Full text available: [pdf\(355.83 KB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: distributed shared memory, distributed virtual environments, object-oriented graphics, shared-data object model

11 Communication optimizations for parallel computing using data access information

Martin C. Rinard

December 1995 **Proceedings of the 1995 ACM/IEEE conference on Supercomputing (CDROM)**

Full text available: [html\(87.34 KB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

12 Pen computing: a technology overview and a vision

André Meyer

July 1995 **ACM SIGCHI Bulletin**, Volume 27 Issue 3

Full text available: [pdf\(5.14 MB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This work gives an overview of a new technology that is attracting growing interest in public as well as in the computer industry itself. The visible difference from other technologies is in the use of a pen or pencil as the primary means of interaction between a user and a machine, picking up the familiar pen and paper interface metaphor. From this follows a set

of consequences that will be analyzed and put into context with other emerging technologies and visions. Starting with a short historic ...

13 Java resources for computer science instruction

Joseph Bergin, Thomas L. Naps, Constance G. Bland, Stephen J. Hartley, Mark A. Holliday, Pamela B. Lawhead, John Lewis, Myles F. McNally, Christopher H. Nevison, Cheng Ng, George J. Pothering, Tommi Teräsvirta

December 1998 **ACM SIGCSE Bulletin**, Volume 30 Issue 4

Full text available:  [pdf\(2.29 MB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

The goal of this working group was to collect, evaluate, and foster the development of resources to serve as components of both new and revised traditional courses that emphasize object-oriented software development using Java. These courses could, for example, integrate Internet-based distributed programming, concurrency, database programming, graphics and visualization, human interface design and object-oriented development. They could therefore also be suitable as capstone courses in computer ...

14 Programming languages for mobile code

Tommy Thorn

September 1997 **ACM Computing Surveys (CSUR)**, Volume 29 Issue 3

Full text available:  [pdf\(393.65 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Sun's announcement of the programming language Java more than anything popularized the notion of mobile code, that is, programs traveling on a heterogeneous network and automatically executing upon arrival at the destination. We describe several classes of mobile code and extract their common characteristics, where security proves to be one of the major concerns. With these characteristics as reference points, we examine six representative languages proposed for mobile code. The conclusion ...

Keywords: Java, Limbo, Objective Caml, Obliq, Safe-Tcl, distribution, formal methods, mobile code, network programming, object orientation, portability, safety, security, telescript

15 Language-level support for exploratory programming of distributed virtual environments

Blair MacIntyre, Steven Feiner

November 1996 **Proceedings of the 9th annual ACM symposium on User interface software and technology**

Full text available:  [pdf\(1.68 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: distributed shared memory, distributed virtual environments, shared-data object model, virtual reality

16 AMTRAM, a remote-terminal, conversational-mode computer system

Juris Reinfelds, L. A. Flenker, R. N. Seitz, P. L. Clem

July 1966 **Communications of the ACM**, Volume 9 Issue 7

Full text available:  [pdf\(654.43 KB\)](#) Additional Information: [full citation](#)

17 The Langley remote computing terminal system: implementation and first year's

operation

Roger V. Butler

July 1966 **Communications of the ACM**, Volume 9 Issue 7

Full text available:  [pdf\(654.43 KB\)](#) Additional Information: [full citation](#)



18 Applications of computer graphics

Joseph Behar

July 1966 **Communications of the ACM**, Volume 9 Issue 7

Full text available:  [pdf\(654.43 KB\)](#) Additional Information: [full citation](#)



19 An interactive remote visualization environment for an electromagnetic scattering simulation on high performance computing system

G. Cheng, Y. Lu, G. Fox, K. Mills, T. Haupt

December 1993 **Proceedings of the 1993 ACM/IEEE conference on Supercomputing**

Full text available:  [pdf\(987.60 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)



20 Programming languages for symbolic numeric and hybrid computation

Christopher J. Shaw

July 1966 **Communications of the ACM**, Volume 9 Issue 7

Full text available:  [pdf\(654.43 KB\)](#) Additional Information: [full citation](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)